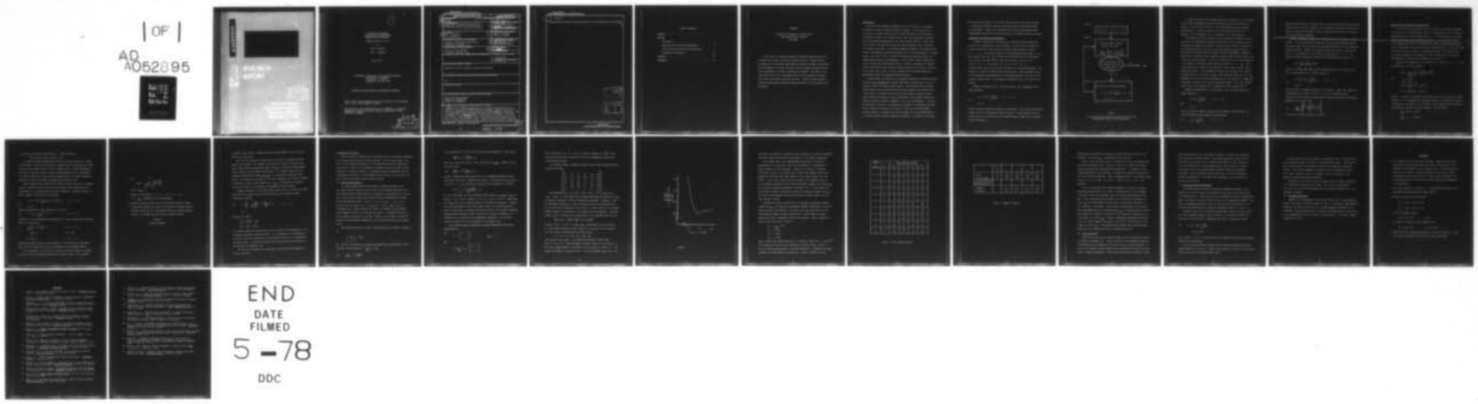
AD-A052 895   FLORIDA UNIV GAINESVILLE DEPT OF INDUSTRIAL AND SYS--ETC   F/G 12/1
COMPUTATION TECHNIQUES FCR LARGE SCALE UNDISCOUNTED MARKOV DECI--ETC(U)
MAR 78   T J HODGSON, G J KOEHLER   N00014-76-C-0096

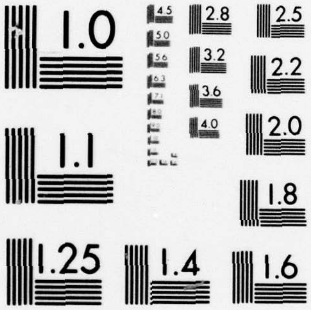
UNCLASSIFIED   RR-78-3   NL

| OF |

AD
A052895



END
DATE
FILMED
5 -78
DDC


1.0
4.5
5.0
5.6
6.3
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6

1

COMPUTATION TECHNIQUES FOR LARGE SCALE UNDISCOUNTED MARKOV DECISION PROCESSES

Research Report No. 78-3

by

Thom J. Hodgson*

Gary J. Koehler**

March, 1978

*Department of Industrial and Systems Engineering
**Department of Management
University of Florida
Gainesville, Florida 32611

*This research was supported in part by the Office of Naval Research, under contract number N00014-76-C-0096.

THE FINDINGS OF THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE NAVY POSITION, UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

14 REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RR-78-3 ✓	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 12 28p.
4. TITLE (and Subtitle) 6 Computation Techniques for Large Scale Undiscounted Markov Decision Processes.		5. TYPE OF REPORT & PERIOD COVERED 9 Research ~~Technical~~ rept.
		6. PERFORMING ORG. REPORT NUMBER 78-3
7. AUTHOR(s) 10 Thom J. Hodgson Gary J. Koehler		8. CONTRACT OR GRANT NUMBER(s) 15 N00014-76-C-0096
9. PERFORMING ORGANIZATION NAME AND ADDRESS Dept Industrial and Systems Engineering ✓ University of Florida Gainesville, Florida 32611		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 16 20061102A14D Rsch in & Appl of Applied Math.
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Arlington, Virginia 22217		12. REPORT DATE 11 Mar[redacted]78
		13. NUMBER OF PAGES 25
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

N/A

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Markov decision processes
Large scale computation
Asymptotic convergence

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In this paper ~~we~~ considers computation techniques associated with the optimization of large scale Markov decision processes. Markov decision processes and ~~the~~ successive approximation procedures ~~of White~~ are described. Then a procedure for scaling continuous time and renewal processes so that they are amenable to the ~~White~~ second procedure is discussed. The effect of the scale factor value on the convergence rate of the procedure and insights into proper scale factor selection are given. Finally, various methods of achieving

next page

404 399

20. (cont'd)

computational efficiency during execution of the optimization are considered.

TABLE OF CONTENTS

ABSTRACT

Computation Techniques for Large Scale Undiscounted Markov Decision Processes

In this paper we consider computation techniques associated with the optimization of large scale Markov decision processes. Markov decision processes and the successive approximation procedure of White are described. Then a procedure for scaling continuous time and renewal processes so that they are amenable to the White procedure is discussed. The effect of the scale factor value on the convergence rate of the procedure and insights into proper scale factor selection are given. Finally, various methods of achieving computational efficiency during execution of the optimization are considered.

Introduction

One of the most powerful modeling tools for the analysis of controlled probabilistic systems is Markov decision processes. If the system can be structured as a Markov process and the control decisions for the system can be defined in terms of the relevant system costs and operational characteristics (transition probabilities), then there exists a wealth of theory that can be used to find the best (least cost, most profitable) set of decisions for operating the system. As with many modeling techniques, real probabilistic systems, when modelled as Markov processes, tend to have large numbers of system states. The result is that for many interesting and important systems, the computational aspects are overwhelming. In most cases, digital simulation is the only viable alternative modelling tool. However, searching for "optimal" control decisions for the system via digital simulation is at best a trial and error effort, and at worst a tedious, expensive, and confusing exercise in experimental design and response surface techniques.

In many cases, the prospects of large scale optimization of Markov decision processes <u>as an alternative</u> to digital simulation are quite good, if one is willing to tackle the computational aspects. In this paper we first review various forms of non-discounted Markov decision processes and transform each to the form of a standard finite state and action Markov decision process. This procedure was explicitly used by Schweitzer [20] for Markov renewal programs and involves choosing a parameter, b, for the transformation. As noted by Schweitzer, the value of b influences the asymptotic convergence rate when White's iterative procedure [25] is used to solve the transformed Markov decision process. We present theoretical insights into the determination of a b which yields the fastest asymptotic convergence. In practice, one cannot easily find this optimal b, so we also present heuristic rules for choosing b. Computational results based upon the heuristics are given which appear quite promising. Finally for completeness we briefly review several other computational techniques used in solving large scale Markov decision processes.

Background and Problem Transformations

Consider a finite state, discrete time, completely ergodic Markov process which is controlled by a decision maker. For each of the N states (i), at each transition of the process, the decision maker chooses an action $k = 1, \ldots, K_i$. This action results in transition probabilities p_{ij}^k, $j = 1, N$, and a reward (cost) q_i^k. p_{ij}^k is defined as the probability that the process, now in state i and under policy k will move to state j over the next transition of the process. q_i^k is defined as the expected reward (cost) over the next transition for operating the system. The problem is to find the optimal action for each state. Here optimality refers to the maximization (minimization) of the expected reward (cost) rate for the process in steady state. This quantity is referred to as g, the gain of the process.

Howard [7] showed that for a given policy set, the simultaneous set of linear equations,

$$(1) \qquad \begin{aligned} v_i + g &= q_i^k + \sum_{j=1}^{N} p_{ij}^k v_j \qquad i = 1, \ldots, N \\ v_N &= 0 \end{aligned}$$

could be solved to compute the gain g of the process. The v_i's are the relative rewards (costs) of starting the process in state i. Howard showed that the optimal gain for the process could be obtained using a simple policy iterative algorithm (Figure 1).

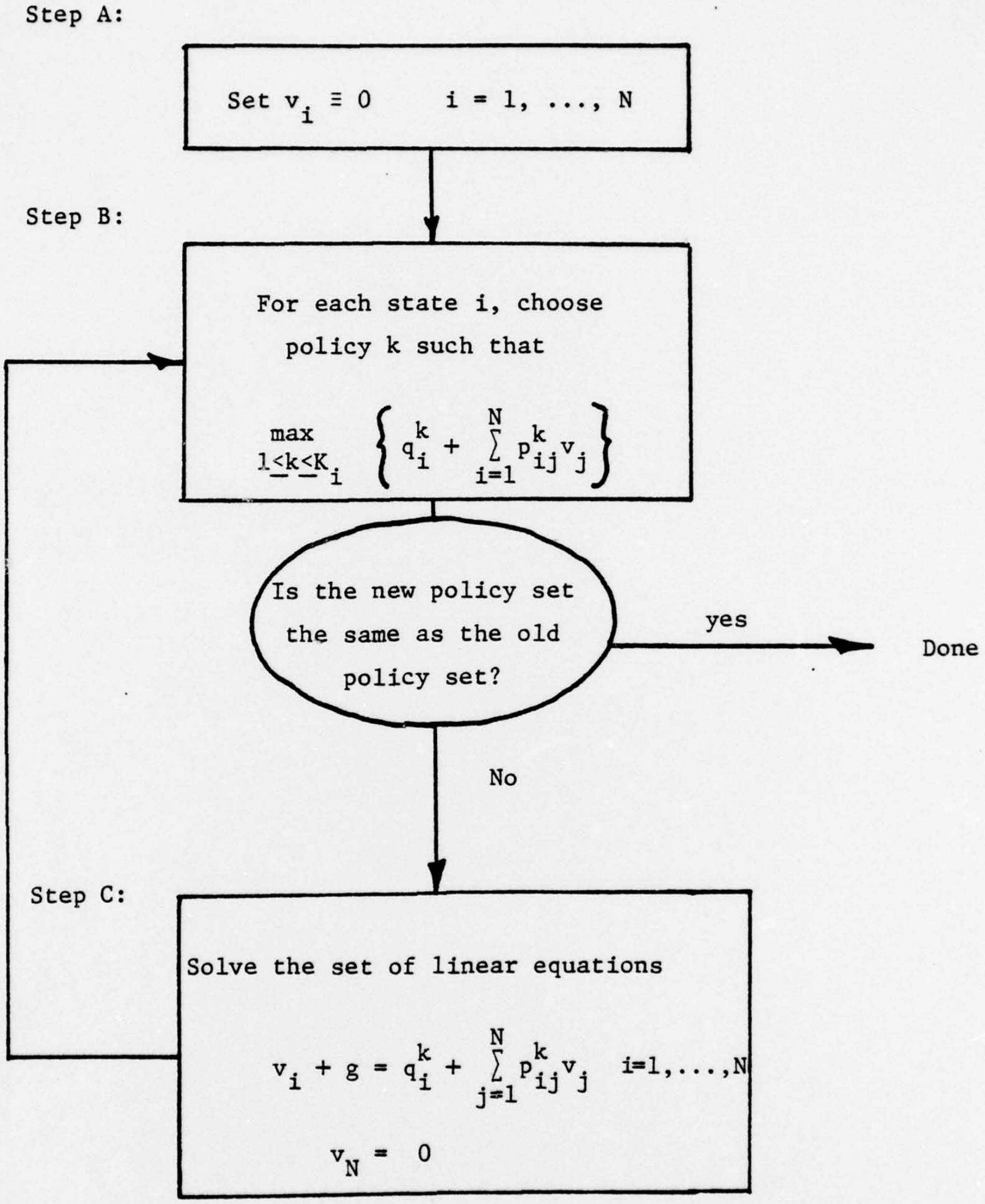

Step A:
Set $v_i \equiv 0 \quad i = 1, \ldots, N$
Step B:
For each state i, choose policy k such that
$\max_{1\le k\le K_i} \left\{ q_i^k + \sum_{i=1}^{N} p_{ij}^k v_j \right\}$
Is the new policy set the same as the old policy set?
yes
Done
No
Step C:
Solve the set of linear equations
$v_i + g = q_i^k + \sum_{j=1}^{N} p_{ij}^k v_j \quad i=1,\ldots,N$
$v_N = 0$

Figure 1

Policy Iterative Procedure for Solving Discrete Time, Undiscounted, Markov Decision Processes.

It should be noted that the Howard algorithm is essentially a dual approach to the linear programming approaches developed by Wolfe and Dantzig [26], Derman [1], Manne [12], and Fox [2] (for Semi Markov Decision Processes). Computationally, the Howard approach is much more efficient than the L.P. approach. As a consequence, the L.P. approach is not considered here.

We now briefly turn our attention to the continuous time Markov decision process and the semi-Markov decision process (which itself subsumes both the continuous and discrete time models as special cases.) However, ultimately we will reduce these latter two cases to the non-discounted Markov decision process and so this diversion is provided only for completeness. Consider a finite state, continuous times, completely ergodic Markov process which is controlled by a decision maker. For each of the N states (i), at each transition of the process, the decision maker chooses an action $k = 1, \ldots, K_i$. This action results in a transition rate a_{ij}^k and a reward (cost) rate $\tilde{q}_i^k$. a_{ij}^k is defined as follows: In an increment of time dt, the process, now in state i and under policy k, will move to state j with probability $a_{ij}^k dt$ $(i \neq j)$. The probability of two or more state transitions is of the order dt^2 or higher and is assumed to be zero if dt is taken sufficiently small. $\tilde{q}_i^k$ is the expected reward (cost) rate incurred over a residence in state i using action k.

Howard [7] showed that for a given policy set, the simultaneous set of linear equations,

$$(2) \qquad \begin{aligned} g &= \tilde{q}_i^k + \sum_{j=1}^{N} a_{ij}^k v_j, \qquad i = 1, \ldots, N, \\ v_N &= 0 \end{aligned}$$

could be solved to compute the gain g of the process. Note that the v_i's are still relative state rewards (costs) and that

$$(3) \qquad a_{ii} = -\sum_{j \neq i} a_{ij}^k, \qquad i = 1, \ldots, N.$$

Howard also showed that the optimal gain for the process could be obtained using a simple policy iterative algorithm. The algorithm is the same as that given in Figure 1, except that equations (2) must be substituted for Step C, and Step B must be appropriately modified.

Finally, consider a finite state, completely ergodic semi Markov decision process. This is essentially the same as the discrete time decision process described earlier in that there is an underlying Markov process with transition probabilities p_{ij}^{k}. However, the holding (transition) time (m) in going from state i to j is described by the density function $h_{ij}^{k}(m)$, $0 < m < \infty$. The expected holding (transition) time, given the system starts in state i is

$$T_i^k = \sum_{j=1}^{N} p_{ij}^{k} \int_0^\infty m h_{ij}^{k}(m)\,dm > 0$$

Jewell [8] showed under rather general assumptions that for a given policy set, the simultaneous set of linear equations,

$$(4) \qquad \begin{aligned} & v_i + T_i^k g = q_i^k + \sum_{j=1}^{N} p_{ij}^{k} v_j , \qquad i = 1, \ldots, N \\ & v_N = 0 \end{aligned}$$

could be solved to compute the gain g of the process. Jewell also showed that the optimal gain for the process could be obtained using a modified version of Howard's simple policy iterative algorithm by substituting equations (4) for Step C and

$$(5) \qquad \max_{1 \leq k \leq K_i} \left\{ \frac{q_i^k + \sum_{i=1}^{N} p_{ij}^{k} v_j - v_i}{T_i^k} \right\}$$

for the test function in Step B of Figure 1.

White's Method and Problem Transformations

It is easy to see from Figure 1 that for each of the processes described that the bulk of the computational effort in the algorithm lies in solving the simultaneous set of linear equations (Step C). For large processes, straight forward techniques, such as Gaussian Elimination, quickly become untenable. In an elegant paper, White [25] proposed a successive approximation approach for the undiscounted, discrete time, Markov decision process. Odoni [16] added bounds for g which are useful in termination decisions. We can also relax the complete ergodicity requirements in line with those given in footnote 1. The White-Odoni technique can be summarized as follows:

Assume we have computed sets of values $V_i(n-1)$, $v_i(n-1)$, $i = 1, \ldots, N$ and a quantity g_{n-1}. We then compute a new set

$$\begin{aligned} V_i(n) &= \max_{1 \leq k \leq K_i} \left\{ q_i^k + \sum_{j=1}^{N} p_{ij}^k v_j(n-1) \right\} , \\ g_n &= V_M(n), \\ v_i(n) &= V_i(n) - g_n, \\ L''(n) &= \max_i \{V_i(n) - v_i(n-1)\} \\ L'(n) &= \min_i \{V_i(n) - v_i(n-1)\} \end{aligned} \tag{6}$$

where M is a state of the process such that for all sets of policies and some integer $u > 0$, the probability of reaching state M in u transitions, starting in any state i, is nonzero for all states i. White showed that the repeated application of equations (6) will converge.[1] In other words,

$$\lim_{n \to \infty} v_i(n) = v_i, \text{ and}$$

$$\lim_{n \to \infty} g_n = g, \text{ where}$$

v_i and g are as deinfed for equations (1). Odoni showed that

$$L''(n) \geqq L''(n+1) \geqq g_n \geqq L'(n+1) \geqq L'(n) .$$

In practice, White's alglorithm has proven to be very effective for large scale systems. The iterative procedure is stable and self-correcting and, since no new data are created (except for the working vectors), storage requirements are fixed. Along these lines, it pays to take advantage of any supersparsity [9] (the vast majority of large scale processes do have very sparse transition matrices) so that the procedure can take place entirely in-core.

While straight forward application of White's approach does not, in general, work for continuous time, and semi-Markov processes, these processes can be transformed to a form compatible with White's approach. Consider equations (2) with v_i added to both sides of the equation.

$$(7)\qquad \begin{aligned} & v_i + g = \tilde{q}_i^k + \sum_{j \neq i} a_{ij}^k v_j + (1 + a_{ii}^k) v_i, \qquad i = 1, \ldots, N, \\ & v_N = 0 \end{aligned}$$

Noting the definition of a_{ii}^k (equation (3)), then if

$$(8)\qquad 0 > a_{ii} = -\sum_{j \neq i} a_{ij}^k > -1, \qquad i = 1, \ldots, N,$$

equation (7) is of the same form as equation (1). This is easily seen by noting that if (7) holds, then

$$\sum_{j \neq i} a_{ij}^k + (1 + a_{ii}^k) = 1 \qquad i = 1, \ldots, N,$$

$$a_{ij}^k \geq 0 \qquad i \neq j, \text{ and}$$

$$1 + a_{ii}^k > 0 \qquad i = 1, \ldots, N.$$

Substituting $1+a_{ii}^k$ for a_{ii}^k in the rate matrix, it is seen that the new matrix $\{a_{ij}^k\}$, for each action set, has all the properties of a stochastic matrix.

As a consequence, if (8) holds, it follows that White's method can indeed be used to solve the continuous time Markov decision process. The procedure

1. Let

$$a_{\max} = \max_{\substack{i=1,\ldots,N \\ k=1,\ldots,K_i}} \left\{ |a_{ii}^k| \right\}$$

(Note: $a_{\max} > 0$)

2. Divide all a_{ij}^k and $\tilde{q}_i^k$, $i, j = 1, \ldots, N$, $k = 1, \ldots, K_i$ by $b > a_{\max}$. Condition (8) is now satisfied.

3. Using the new a_{ij}^k and $\tilde{q}_i^k$, solve the problem using White's method.

4. To express the results in terms of the original continuous process, multiply the gain g by b. The optimal policy and relative rewards (costs), v_i, obtained are valid for the original process.

Figure 2

Scaling Procedure

in Figure 2 will convert a suitable continuous time problem so that the consitions of (8) will hold.

Note that the scaling of the problem really amounts to changing the time frame of the problem. For instance, if the process is stated in terms of per minute (a_{ij}^k) and dollars per minute ($\tilde{q}_i^k$), and $a_{max} = 60$ then the transformation simply converts the time frame to hour units. It is readily seen that it is necessary to divide by at least a_{max}($a_{max} > 1$) to end with a stochastic $\{a_{ij}^k\}$ matrix. The question of interest is: Can the convergence rate of White's method be improved by using a proper choice of $b > a_{max}$? We consider that question shortly; but first, let us address the semi-Markov decision process.

Consider equations (4) with the relative reward (cost) v_i moved to the right hand side of the equation and both sides divided by the expected holding (transition) time T_i^k.

$$(9) \qquad g = \frac{q_i^k}{T_i^k} + \sum_{j \neq i} \frac{p_{ij}^k v_j}{T_i^k} + \frac{(p_{ii}^k - 1)}{T_i^k} v_i \qquad i = 1, \ldots, N,$$

$$v_N = 0$$

Letting $\tilde{q}_i^k = a_i^k / T_i^k$,

$$a_{ij}^k = p_{ij}^k / T_i^k \text{ , and}$$

$$a_{ii}^k = (p_{ii}^k - 1)/T_i^k \text{ ,}$$

it is readily seen that equations (9) are of the same form as equations (2), the continuous time Markov decision process. As a consequence, the transformation can also be applied to equations (9) to facilitate solution of the semi-Markov decision process. It should be noted that the transformation is equivalent to one developed by Schweitzer [20].

We now turn our attention to the problem of speeding the convergence of White's algorithm.

Convergence Facilitation

There are several procedures that have been used in accelerating convergence in solving discounted Markov decision processes. By and large, though, these have not been examined extensively in the non-discounted Markov decision process context. Briefly, the acceleration techniques include (a) problem transformation, (b) cheap iterations, (c) suboptimal activity elimination, and (d) extrapolation procedures. We will discuss each of these, in turn, in the context of the non-discounted Markov decision process.

(a) Problem Transformation

In solving (generalized)discounted Markov decision processes, it is well known that the largest spectral radius of the transition matrices (i.e., the process spectral radius) governs the asymptotic convergence rate. Porteus [18], Totten [24] and others have devised problem transformations to reduce the process spectral radius. Morton and Wecker [14] have shown that asymptotic relative values and policy convergence are at least of order $(\alpha\lambda)^n$ where λ is greater than the subdominant eigenvalue[2] and $0<\alpha<\infty$ is the discount factor. A reasonable question to ask is whether the choice of b in Step 2 (Figure 2) can be made to reduce the modulus of the subdominant eigenvalue of the transition matrix of the optimal policy.

The transition matrix for policy δ resulting from the procedure of Figure 2 is

$$I + \frac{1}{b} A_\delta \text{ , where}$$

$$A_\delta = P_\delta - I.$$

Let λ and $\bar{x}$ be an eigenvalue and associated eigenvector, respectively, of the starting transition matrix $I + \frac{1}{a_{max}} A_\delta$. Then

$$(10) \qquad \frac{a_{max}}{b} \lambda + \frac{b - a_{max}}{b}$$

is an eigenvalue of $I + \frac{1}{b} A_\delta$ with $\bar{x}$ its associated eigenvector. Now clearly

$$\frac{a_{max}}{b} \, re\lambda + \frac{b - a_{max}}{b} \geqq re\lambda$$

where $re\lambda$ is the real part of λ with $-1 \leqq re\lambda \leqq 1$ and $b > a_{max} \geqq 0$. However, it may not be true that

$$(11) \qquad \left| \frac{a_{max}}{b} \lambda + \frac{b - a_{max}}{b} \right| \geqq |\lambda| \; .$$

Suppose δ indexes an optimal policy and λ is a subdominant eigenvalue associated with this policy. Expanding the square of the modulus of both sides of (11) with $\lambda = \lambda_1 + \lambda_2 i$ gives that a reduction in the modulus of λ requires

$$(1-\lambda_1) \left[\lambda_1 + \frac{b - a_{max}}{a_{max} + b} \right] \leqq \lambda_2^{\,2} \; .$$

If $\lambda_2 = 0$, then either $\lambda_1 = 1$ and no reduction can be made or $\lambda_1 \leqq (a_{max} - b)/(a_{max} + b)$ and λ_1 is necessary negative. In this case, it would appear that any $b > a_{max}$ will yield a resultant benefit in asymptotic convergence. However, this is not necessarily true, since we may "bump" into another eigenvalue. That is, increasing b to decrease the absolute value of the dominant (negative) eigenvalue will eventually result in some other (positive) eigenvalue increasing until it becomes the new subdominant eigenvalue. At that point further increases in b will not improve the convergence rate.

The following example illustrates an extreme improvement from problem transformation.

Let

$$A_\delta = \begin{pmatrix} -.3 & .3 \\ .5 & -.5 \end{pmatrix} \qquad\qquad a_{max} = .5$$

and

$$I + \frac{1}{a_{max}} A_\delta = \begin{pmatrix} .4 & .6 \\ 1 & 0 \end{pmatrix}$$

with a spectrum of {1, -.6}. For b>.5 we have a spectrum $\{1, \frac{b-.8}{b}\}$. Here we want b=.8 which gives a modulus of 0.0 for the subdominant eigenvalue of the transformed process.

As a further example, consider the Markov process whose transition matrix is given as follows:

$$\begin{bmatrix} .31 & .13 & .21 & .05 & .10 & .20 \\ .15 & .12 & .16 & .20 & .12 & .25 \\ .02 & .01 & .01 & .01 & .93 & .02 \\ .12 & .28 & .09 & .16 & .04 & .31 \\ 0 & .01 & .85 & 0 & .09 & .05 \\ .11 & .30 & .10 & .15 & .14 & .20 \end{bmatrix}.$$

The eigenvalues are 1.0, -.8421, .6945, .2079, -.085 + .0116i, and -.085 - .0116i. It would appear that problem transformation should be of value in speeding convergence, since the subdominant eigenvalue is negative. From the preceding development, it would be expected that the convergence rate of the process would be maximized at the value of "b" which results in the largest negative eigenvalue being equal to the largest positive eigenvalue. Applying equation (10), to equate the two eigenvalues of the tranformed matrix, we get

$$\frac{.99}{b}(.8421) - \frac{b-.99}{b} = \frac{.99}{b}(.6945) + \frac{b-.99}{b}$$

Solving, we get b = 1.063. In other words, transforming the process using b = 1.063 should achieve the "best" asymptotic convergence for the process. As a test, White's Algorithm was run using costs of

$$\bar{q} = (1.14, 2.27, 5.06, 2.97, 3.96, 4.90)$$

(only one policy per state). The problem was declared "solved" when $L''(n) - L'(n) \leq 10^{-4}$. Runs were made for various values of b (see Figure 3). The actual minimum number of iterations (30) occurred for a value of b $\approx$ 1.09, whereas the number of iterations for b $\approx$ 1.063 was slightly higher (31). The

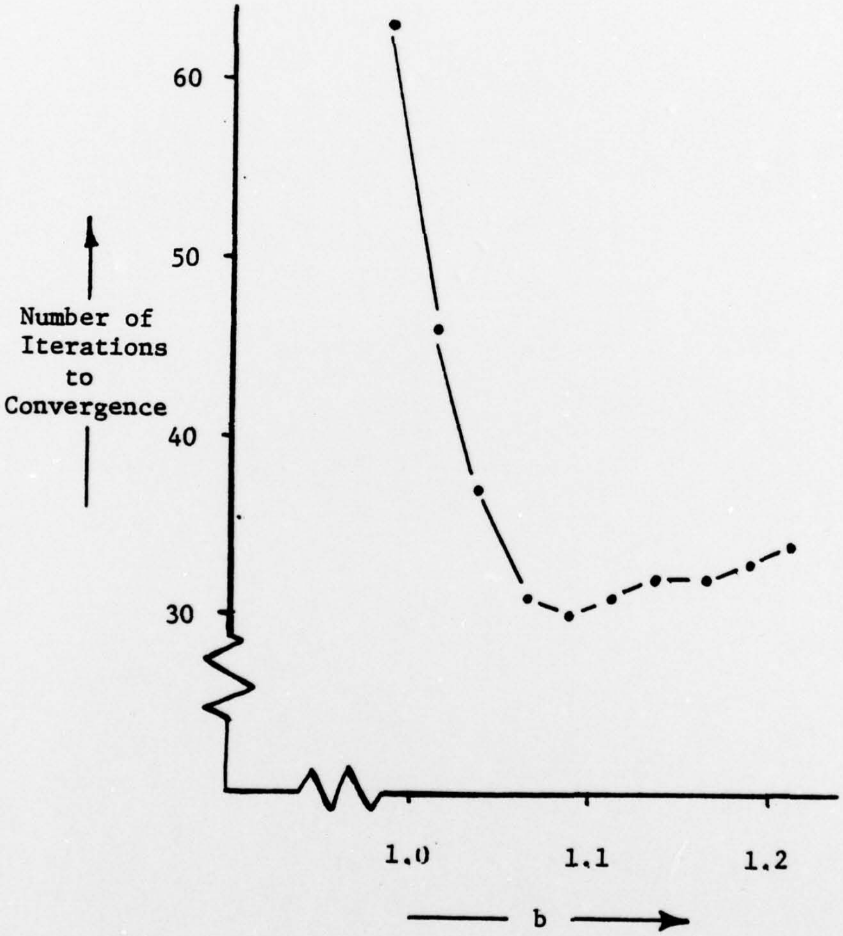


Figure 3

inaccuracy in prediction is expected, since the method of prediction considers only main effects and ignores the contribution of the smaller eigenvalues.

As one might expect, the straightforward application of the above observations is not practical, since the determination of eigenvalues for large processes is itself difficult. However, in practice it is usually intuitively obvious to the analyst that a process may possess strong cyclic tendencies, indicating that some eigenvalue has a large negative real component. If the cyclic tendency is strong enough, this eigenvalue will be the subdominant eigenvalue and the above development suggests that some $b>a_{max}$ may decrease the resulting asymptotic convergence rate. In any event, applying White's method, using several values of b marginally larger than a_{max}, and noting the convergence rate of the process for various values of b can many times be of value.

In testing the above we noted that if b was made successively slightly larger than a_{max} that either the convergence improved dramatically or the convergence slightly deteriorated. To further test this observation, we randomly generated Markov decision problems with a varied number of states. Within each state ten different actions were available. White's method was used to solve each using b values of

$$\begin{aligned} b_0 &= a_{max} + 10^{-5} \\ b_1 &= 1.05b_0 \\ b_2 &= 1.10b_0 \\ b_3 &= 1.15b_0 \end{aligned}$$

Again, problems were declared "solved" at iteration n when $L''(n) - L'(n) \leq 10^{-4}$. If a problem was solved in fewer iterations for some b_i than b_j with $i>j$, then the problem transformation was declared beneficial. Otherwise the transformation was classified as non-beneficial. Clearly a problem could be

NUMBER OF STATES	N_n N_b	TOTAL ITERATION COUNTS			
		b_0	b_1	b_2	b_3
3	6	67	73	79	85
	4	79	61	51	43
4	7	159	166	175	183
	2	51	43	36	31
5	7	150	157	166	174
	3	49	42	40	40
6	2	34	36	38	40
	8	212	160	141	133
7	4	60	63	67	71
	5	95	82	77	76
8	7	114	121	129	136
	8	185	145	127	120
9	3	50	52	56	58
	11	430	260	210	193
10	6	134	139	146	152
	10	313	238	212	199
15	2	63	67	70	74
	8	692	277	222	217
20	2	65	68	71	74
	8	266	212	194	189

Table 1: Total Iteration Counts

	N_n				
	N_b	b_0	b_1	b_2	b_3
Total	46	896	942	997	1047
Across States	67	2372	1520	1310	1241
▨					
Problem		19.5	20.5	21.7	22.8
Averages		35.4	22.7	19.6	18.5

Table 2: Summary of Table 1.

mislabelled as non-beneficial using the grid given above but may in fact be beneficial for some $b>a_{max}$. The opposite is not the case.

Table 1 gives the total number of iterations to solve the non-beneficial and beneficial problem cases. N_n and N_b stand for the number of problems labelled non-beneficial and beneficial, respectively. For example, in the randomly generated 8 state problems, 7 problems were labelled non-beneficial and 8 were labelled beneficial. Table 2 summarizes Table 1 by providing totals across states and then averages across states and over problems.

If we can assume that the average performance of the set of randomly generated problems used in this study is representative of the performance of the set of real world problems, then the following observations can be made. First, problems whose convergence can be improved by increases in b above a_{max} are those problems that are hard to solve anyway (see Table 2, 19.5 versus 35.4 iterations). Second, when a problem does not show convergence improvement when b is increased above a_{max}, the deterioration in convergence speed is not dramatic (see Table 2, 19.5 versus 22.8 iterations for a 15% increase in b above a_{max}). Finally, convergence improvements, when they occur, are rather dramatic (see Table 2, 35.4 to 18.5 iterations for a 15% change in b above a_{max}). These observations suggest using problem transformation can be of significant value in speeding convergence.

(b) Cheap Iterations

Cheap iterations were first noted by Morton [13] and discussed in detail by Zaldivar and Hodgson [26]. "Cheap Iterations" are accomplished simply by not performing policy maximization at every iteration of White's method. If one does not perform a policy maximization the computational effort per iteration is reduced considerably. This approach makes sense intuitively in that there are both policy sets, and relative values (v_i) and gain (g) converging in the operation of White's method. Using cheap iterations allows the relative values (v_i) and gain (g) to converge sufficiently so that a new policy set can be chosen which is significantly better than the old one. Clearly, in practice, there is an optimum tradeoff between "cheap" and "expensive" iterations. In our experience, we have used from 5 to 30 cheap iterations per policy maximization. The number being dependent on the convergence properties of the process.

(c) Suboptimal Activity Elimination

An extremely useful procedure in dynamic programming methodology is the reduction of the policy space by determining actions that could never be part of an optimal policy. These actions can be eliminated from further consideration. Hence, the problem can actually shrink in size as the computations progress. The idea of eliminating suboptimal activities was first given by MacQueen [11] in the discounted Markov decision context and refined by others [4, 5, 18, 24].

The basic idea, cast in the non-discounted Markov decision process context, is to first determine bounds on $\binom{v}{g}$ at iteration n (call them ℓ^n and u^n) and then test, for each activity k associated with state i, whether the system

$$(12) \qquad g + v_i = q_i^k + \sum_{j=1} p_{ij}^k v_j$$

$$\ell^n \leq \binom{v}{g} \leq u^n$$

has a solution. If not, k cannot be part of an optimal policy and can be removed from further consideration.

For the discounted Markov decision process, several researchers have presented bounds [10, 18, 19, 24]. However, even though $w^n \to w$ and $g^n \to g$, no bounds have been given for the non-discounted case.[3]

Recently Hastings [3] has proposed a suboptimality test. His test identifies non-optimal actions for state i at value iteration stage n. This does not mean that the detected "non-optimal" actions are non-optimal for subsequent stages.[4] Thus, actions flagged at iteration n must be re-examined at some later stage. Hasting's test can be thought of as an "intelligent" intermediate to expensive and cheap iterations.

We might note here that any type of relaxed iteration (cheap or Hasting's) will invalidate the bounds given by Odoni. That is, to be valid bounds, L"(n) and L'(n) can only be determined from the unrelaxed iteration (i.e., the expensive iteration).

(d) Extrapolation Methods

Generally, the convergence of the relative values (v_i) to their respective values takes place in an orderly fashion so that it is possible to make educated guesses at the final values of the v_i's, thereby speeding convergence of the algorithm. Simple approaches such as linearly extrapolating each of the trends of the progressions of the v_i's seem to be most effective. For a more complete discussion, see [27].

FOOTNOTES

1. The assumptions used by White can be relaxed. Schweitzer [22] proved convergence for the general single chain acyclic process while Su and Deininger [23] extended this to the periodic case. Such conditions are hard to test in practice. Recently Platzman [17] has given a weaker condition that can be readily tested. Finally, Morton and Wecker [14] have generalized most of the above plus have added some new dimensions to the algorithm.

2. The largest eigenvalue is always 1.0. The subdominant eigenvalue is the remaining eigenvalue having the largest modulus.

3. We warn the reader that the "bounds"

$$u_i^n = V_i(n) + L''(n) - L'(n)$$

$$\ell_i^n = V_i(n) - L''(n) + L'(n)$$

do not (in general) satisfy

$$u_i^n \geqq V_i(n) \geqq \ell_i^n \qquad m \geqq n.$$

All that can be shown for these bounds is that

$$u_i^n \geqq V_i(m) \geqq \ell_i^n \qquad n-1 \leqq m \leqq n+1.$$

4. Under fairly mild conditions, Hastings [3] shows that there is a stage after which non-optimal actions will be properly identified.

REFERENCES

1. Derman, C., "On Sequential Decisions and Markov Chains," Management Science, 9, No. 1, 1962-1963, pp. 16-24.

2. Fox, B. L., "Markov Renewal Programming by Linear Fractional Programming," SIAM Journal Applied Math., 14, 1966, pp. 1418-1432.

3. Hastings, N. A. J., "A Test for Non-Optimal Actions in Undiscounted Finite Markov Decision Chains," Management Science, 23, No. 1, 1976, pp. 87-92.

4. Hastings, N.A.J. and J.M.C. Mello, "Erratum Tests for Suboptimal Actions in Discounted Markov Programming," Management Science, 20, No. 17, 1974, p. 1143.

5. Hastings, N.A.J. and J. M. C. Mello, "Tests for Suboptimal Actions in Discounted Markov Programming," Management Science, 19, No. 9, 1973, pp. 1019-1022.

6. Hordijk, A. and H. Tijms, "The Method of Successive Approximations and Markovian Decision Problems," Operations Research, 22, 1974, pp. 519-521.

7. Howard, R. A., Dynamic Programming and Markov Processes, MIT Press and Wiley, New York, 1960.

8. Jewell, W. S., "Markov Renewal Programming: I and II," ORSA, 11, Nov.-Dec., 1963, pp. 938-971.

9. Kalan, J. E., "Aspects of Large-Scale, In-Core Linear Programming," Proceedings of ACM Annual Conference, Chicago, Illinois, August 3-5, 1971.

10. MacQueen, J., "A Modified Dynamic Programming Method for Markovian Decision Problems," J. Mathematical Analysis and Appl., 14, 1966, pp. 38-43.

11. MacQueen, J. B., "A Test for Suboptimal Actions in Markovian Decision Problems," Operations Research, 15, 1967, pp. 559-561.

12. Manne, A. S., "Linear Programming and Sequential Decisions," Management Science, 6, 1960, pp. 259-267.

13. Morton, T. E., "On the Asymptotic Convergence Rate of Cost Differences for Markovian Decision Processes," Operations Research, 19, 1971, pp. 244-248.

14. Morton, T. E. and W. E. Wecker, "Discounting, Ergodicity, and Convergence for Markov Decision Processes," Management Science, 23, 1977, pp. 890-900.

15. Nering, E. D., Linear Algebra and Matrix Theory, 2nd. Ed., John Wiley and Sons, New York, 1970.

16. Odoni, A. R., "On Finding the Maximal Gain for Markov Decision Processes," Operations Research, 17, 1969, pp. 857-860.

17. Platzman, L., "Improved Conditions for Convergence in Undiscounted Markov Renewal Programming," Operations Research, 25, No. 3, 1977, pp. 529-533.

18. Porteus, E. L., "Bounds and Transformations for Discounted Finite Markov Decision Chains," Operations Research, 23, No. 4, 1975, pp. 761-784.

19. Porteus, E. L., "Some Bounds for Discounted Sequential Decision Processes," Management Science, 18, No. 1, 1971, pp. 7-11.

20. Schweitzer, P. J., "Iterative Solution of the Functional Equations of Undiscounted Markov Renewal Programming," J. Math. Analysis and Appl., 34, 1971, pp. 495-501.

21. Schweitzer, P. J., "Multiple Policy Improvements in Undiscounted Markov Renewal Programs," ORSA, 19, May-June, 1971, pp. 784-793.

22. Schweitzer, P. J., "Perturbation Theory and Markovian Decision Processes," MIT Operations Research Technical Report, 15, June 1965.

23. Su, S. Y. and R. A. Deininger, "Generalization of White's Method of Successive Approximations to Periodic Markovian Decision Processes," Operations Research, 20, No. 2, 1972, pp. 318-326.

24. Totten, J. C., "Computational Methods for Finite State Finite Valued Markovian Decision Problems," Operations Research Center, University of California, Berkeley, ORC-71, 1971.

25. White, D. J., "Dynamic Programming, Markov Chains, and the Method of Successive Approximations," Journal of Mathematical Analyses and Applications, 6, 1963, pp. 373-376.

26. Wolfe, P. and G. Dantzig, "Linear Programming in a Markov Chain," ORSA, 10, Sept.-Oct., 1962, pp. 702-710.

27. Zaldivar, M. and T. J. Hodgson, "Rapid Convergence Techniques for Markov Decision Processes," Decision Sciences, 6, 1975, pp. 14-24.